

AMENDMENTS TO THE CLAIMS

Please replace all prior versions of the claims with the following claim listing:

Claims:

1. (Original) An active data type for use in a computer program, the active data type being embodied in a computer-readable medium, the active data type comprising:
- an identifier identifying an instance of the active data type, the computer program with which the active data type is utilized identifying the active data type instance by the identifier associated with the active data type instance; and
 - at least a first algorithm associated with the active data type, the first algorithm being configured to be automatically executed when an attempt is made to access a value associated with the active data type instance.
2. (Original) The active data type of claim 1, wherein the attempt to access the value associated with the active data type instance is made by a particular routine, the algorithm automatically determining the value associated with the active data type instance before the routine accesses the value.
3. (Original) The active data type of claim 2, wherein processing by the algorithm is suspended until the particular routine is finished running.
4. (Original) The active data type of claim 3, further comprising:
- a second algorithm associated with the active data type, the second algorithm being configured to be automatically executed once the value associated with the active data type instance has been set, the second algorithm processing the set value to generate value processing results.
5. (Original) The active data type of claim 4, wherein the setting of the value and the processing of the value by the second algorithm is delayed until the particular routine returns, wherein once the particular routine returns, the second algorithm processes the set value to generate value processing results.

6. (Original) The active data type of claim 1, wherein the active data type is a parameter being utilized by the computer program.

7. (Original) The active data type of claim 1, wherein the active data type is a symbol being utilized by the computer program.

8. (Original) The active data type of claim 1, wherein the active data type is a string format being utilized by the computer program, the string format specifying a format of a string data type instance associated with the string format, the string format including a format operation, the format operation specifying an operation associated with the string data type instance.

9. (Currently Amended) The active data type of claim 2, wherein the particular routine is a test routine utilized for testing a device and obtaining measurement results relating to one or more tests performed in testing the device, the test routine being invoked by said computer program, said computer program being a Test Executive computer program for verifying the integrity of electrical circuits being implemented in a Test Executive program environment.

10. (Original) An apparatus for executing a computer program, the apparatus comprising:

first logic configured to execute the computer program and any routines invoked by the computer program, the computer program utilizing at least one active data type, the active data type having an identifier and at least a first algorithm associated therewith, the identifier identifying an instance of the active data type, the first algorithm being automatically executed when an attempt is made to access a value associated with the active data type instance.

11. (Original) The apparatus of claim 10, wherein the attempt to access the value associated with the active data type instance is made by a particular routine, and wherein the first algorithm automatically determines the value associated with the active data type instance before the particular routine accesses the value.

12. (Original) The apparatus of claim 11, wherein once the algorithm has determined the value of the active data type instance, processing by the algorithm is suspended until the routine is finished running.

13. (Original) The apparatus of claim 12, wherein the active data type further has a second algorithm associated therewith, the second algorithm configured to be automatically executed once the value associated with the active data type instance has been set, the second algorithm processing the set value to generate value processing results.

14. (Original) The apparatus of claim 13, wherein the setting of the value and the processing of the set value by the second algorithm is delayed until the particular routine returns, and wherein once the particular routine returns, the second algorithm processes the set value to generate value processing results.

15. (Currently Amended) The apparatus of claim 10, wherein the particular routine is a test routine utilized for testing a device and obtaining measurement results relating to one or more tests performed in testing the device, the test routine being invoked by said computer program, said computer program being a Test Executive computer program for verifying the integrity of electrical circuits being implemented in a Test Executive program environment.

16. (Original) A method for utilizing an active data type in a computer program, the active data type comprising the steps of:

identifying an instance of the active data type with an identifier, the active data type instance being identified by the computer program with which the active data type is utilized; and

automatically executing a first algorithm when an attempt is made to access a value associated with the active data type instance.

17. (Original) The method of claim 16, wherein when the first algorithm is executed, the first algorithm automatically determines the value associated with the active data type instance.

18. (Original) The method of claim 17, wherein the attempt to access the value of the active data type instance is made by a particular routine, and wherein once the first algorithm has determined the value of the active data type instance, processing by the first algorithm is suspended until the particular routine is finished running.

B1
Back.

19. (Original) The method of claim 18, further comprising the step of: automatically executing a second algorithm associated with the active data type when a value associated with the active data type instance is set, the second algorithm processing the set value to generate value processing results.

20. (Original) The method of claim 19, wherein processing by the second algorithm of the set value associated with the active data type is delayed until the particular routine returns.
